



# A0597203 AI Business Applications

## Introduction to Agentic AI

<https://www.knime.com/events/data-aware-agentic-ai-getting-started-course>

## Agentic AI

- Agentic AI is an **AI system that can plan, decide, and act** to achieve goals, rather than just respond passively to prompts.
- It combines LLMs with tools, memory, and reasoning capabilities to perform tasks autonomously.

### Key Characteristics

- **Goal-driven:** Works toward objectives instead of isolated answers.
- **Tool-using:** Can call external functions, databases, or APIs to gather information or take action.
- **Memory-aware:** Retains past interactions and context across steps.
- **Reasoning loop:** Plans multi-step actions, executes them, evaluates results, and adapts.

### Why it matters

- Moves beyond Q&A into **problem-solving agents**.
- Enables automation of workflows, research, customer support, and decision-making.
- Foundation for building **personal AI assistants** and **autonomous applications**.

# Agentic AI Scenario - Travel Assistant

**Task:** Arrange a budget-friendly trip.

## Steps the Agent Takes

### 1. Understand the goal

- User: *"Find me the cheapest way to travel to Paris next weekend, plus affordable lodging and a rental car."*

### 2. Plan the workflow

- Search for flights and compare fares.
- Find nearby hotels with availability and within budget.
- Check for rental cars near the hotel.
- Combine results into a single travel plan.

### 3. Use external tools

- Flight search API → retrieves cheapest available flight.
- Hotel booking API → filters for price, location, and vacancy.
- Car rental service API → checks for affordable, available options.

### 4. Evaluate and optimize

- Ensures hotel is close to airport or city center.
- Chooses options that minimize total cost.
- Cross-checks dates and availability.

### 5. Report and act

- Returns the best itinerary to the user.
- Optionally books the flight, hotel, and car.
- Provides confirmation details in one place.

**Outcome:** The agent delivers a **complete, optimized travel plan** instead of just answering one query.

## Agentic AI

- Agentic AI accomplishes this by combining and orchestrating different tools to complete a required task.
- Based on the input from the user, an AI agent identifies necessary tools, combines them in a correct order, monitors their executions, mediates their inputs and outputs, and formulates the final outcome the user can understand.

## Building Blocks of Agentic AI

- Agentic AI consists of a collection of **tools** specialized in different tasks, and an **agent** that calls them to accomplish the request by the user.
- The agent accesses the tools from a repository, combining them dynamically to accomplish the required process.
- In KNIME Analytics Platform, tools are workflows specialized for designated tasks.
- Such tools can be a modified version of an existing workflow, or constructed from scratch.
- These tools have the ability to receive input from an agent and return the output to the agent.
- A tool may be able to accomplish a particular task with or without using AI.

## What are Tools in KNIME?

- **Tools in KNIME are workflows** specialized in different tasks.
- Tools can be called by an agent with a mechanism to receive input from and to return output to an agent.
- We have two classes of tools: classic tools and intelligent tools:
  - Classic Tools
    - Perform a designated task without using AI
    - Examples: data aggregation, classification, numeric prediction, data transformation
  - Intelligent Tools
    - Perform a designated task using AI models
    - Examples: summarizing document, machine translation, sentiment analysis

Tool

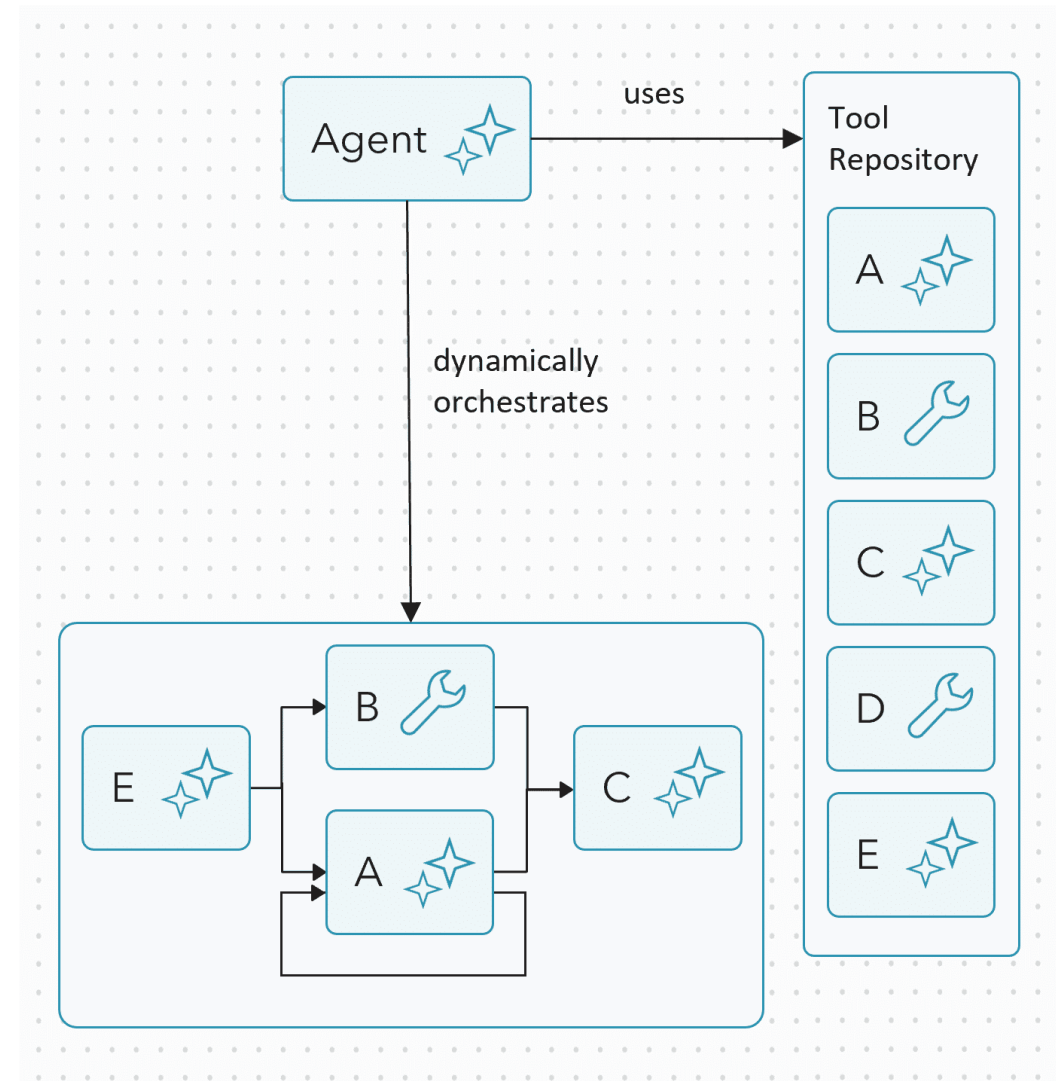


Tool



## What is an Agent?

- An agent is an autonomous software system that can perceive its environment, reasoning about it, making decisions, and taking actions to achieve specific goals.
- It can combine tools at its disposal to achieve the goal to produce the desired outcome.



## Understanding Properties of Tools

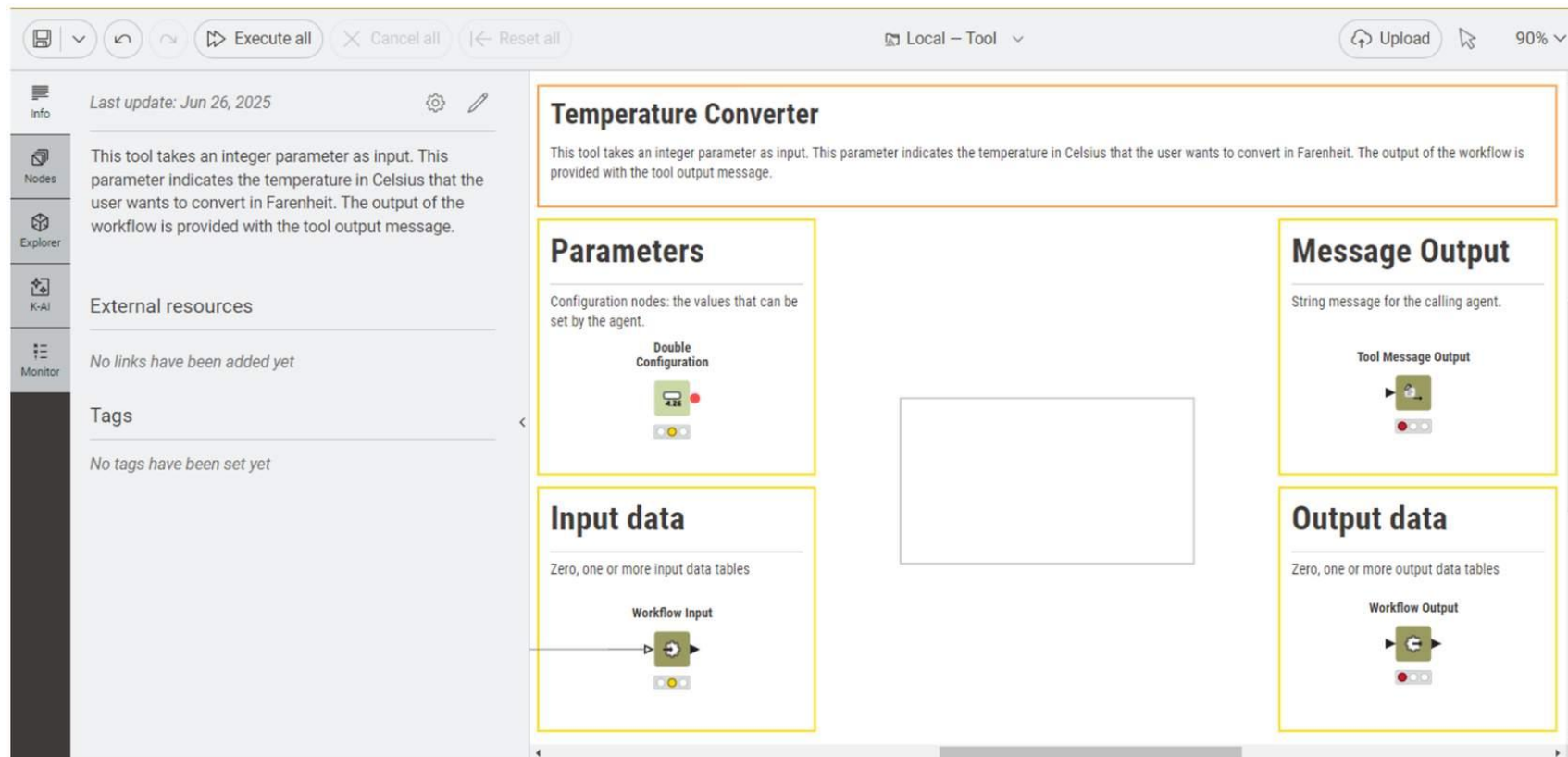
- In KNIME analytics platform, a tool is a workflow that the agent can call to perform a specific task during its reasoning process.
- Tools provide the agent with functionality it can use while thinking step-by-step about how to solve a problem.
- Tools can optionally receive data input from an agent or return a data output to an agent.
- In the following, we introduce two properties tools can have:
  - **Data-aware** tools are tools that access data or perform tasks on some data they receive from an agent. This can be, for example, a Tool that accesses and aggregates customer data from different data sources.
  - **Intelligent** tools are making use of Agentic AI. This can be, for example, a Tool that translates or summarizes a document or a Tool that predicts the sentiment of a message by means of calling a LLM.



# Tools in KNIME

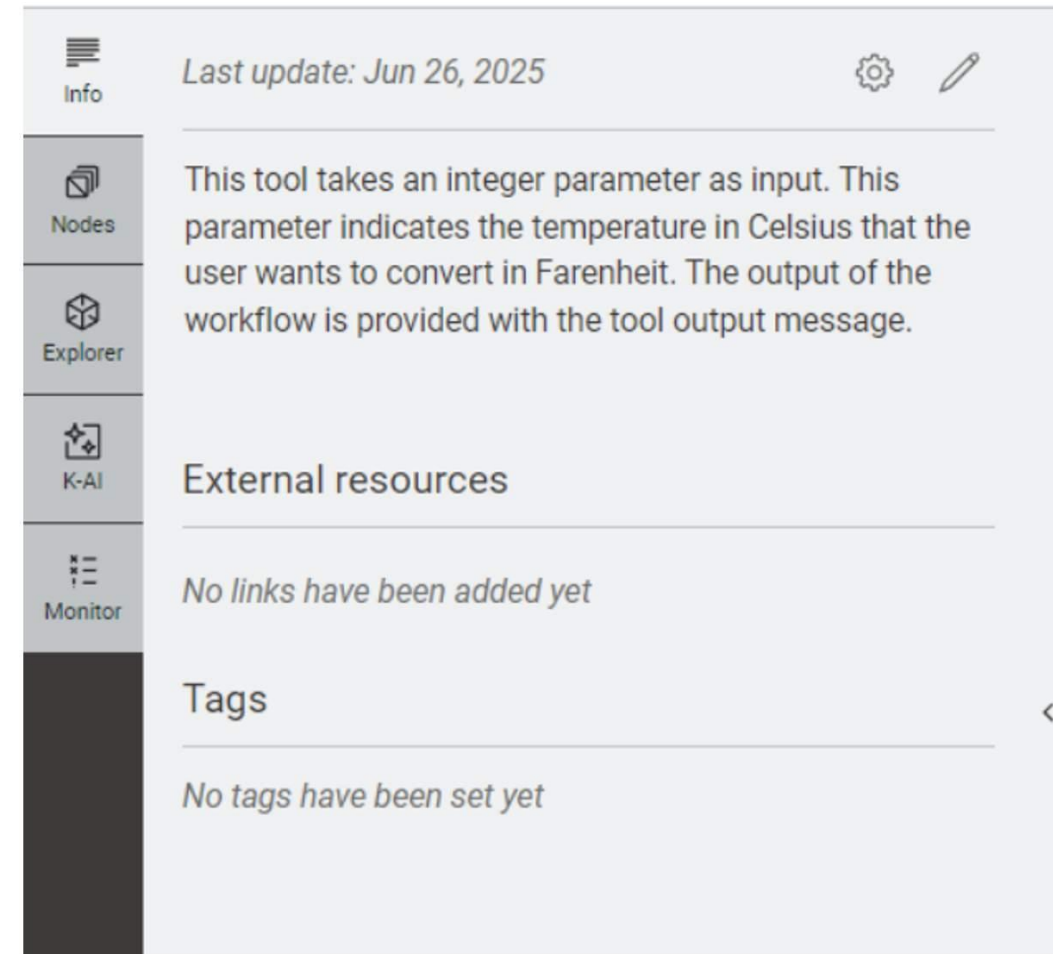
A tool in KNIME is a KNIME workflow that performs a specific task.

However, the tool workflow in KNIME needs to contain certain elements that are required to make the workflow a Tool usable by the agent



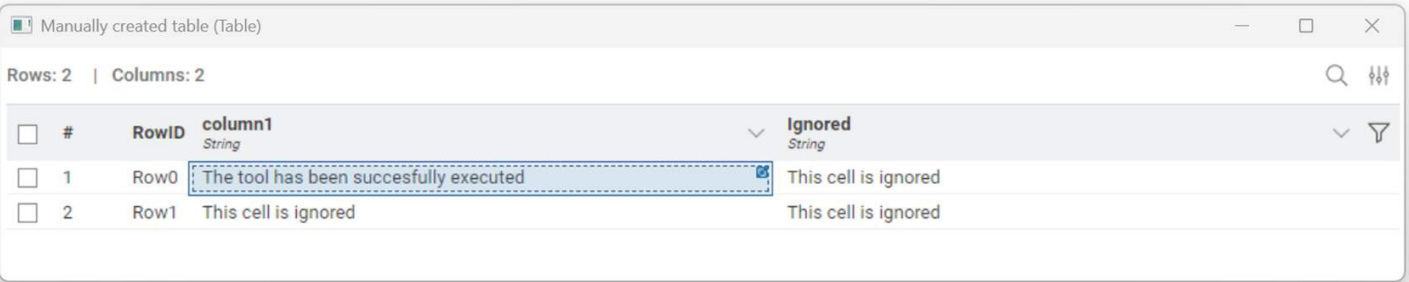
## 1-Tool Description

- Tool description explains as precisely as possible what task the tool is able to solve.
- It is defined in the workflow's Info field.
- The agent reads this description and decides when to use the tool.
- A well-written description allows the agent to reason effectively about the available options.
- We can also include some examples of usage, to enhance the description.
- The description should explain:
  - The task performed by the tool
  - The expected input data
  - The output produced
  - The configurations required
  - The types of question the tool is designed to answer



## 2-Tool Message Output

- The Tool Message Output node provides optional feedback to the agent after a tool execution.
- We can include this node if textual output is needed for the agent to reason with after the tool call (remove it if no output is required from the tool).
- The node reads the first value from the first cell of its input table.
- This string becomes the content of the Tool Message returned to the agent. It is useful, for example, to return
  - A summary of the processed data (“The database contains data about 22 customers.”)
  - Short textual insights (“The average temperature is 22,5°C which is equivalent to 72.5°F.”)
  - Confirmation or intermediate results (“The email has been successfully sent to the recipient list.”)



#	RowID	column1	Ignored
1	Row0	The tool has been successfully executed	This cell is ignored
2	Row1	This cell is ignored	This cell is ignored



## 3-Parameters

- Parameters are values that can be set by the agent.
- We can use configuration nodes (such as String Configuration or Integer Configuration) to define adjustable parameters.
- For each parameter we need to:
  - Provide a clear parameter name used as the variable name
  - Write a concise description explaining its purpose.
- The agent reads these definitions to determine which parameter values to set during tool execution

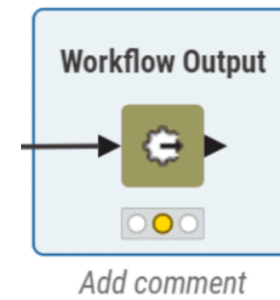
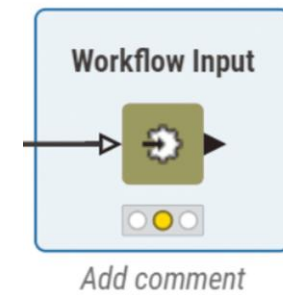
The screenshot shows a dialog box titled "Dialog - 3:1 - Double Configuration". It has three tabs: "Control", "Flow Variables", and "Job Manager Selection". The "Control" tab is selected. The dialog contains the following fields:

- Label:** "Temperature in Celsius"
- Description:** "This parameter indicates a temperature that the user wants to convert from Celsius into Farenheit."
- Parameter/Variable Name:** "temperature\_C"
- Minimum:** A checkbox is unchecked, followed by a text box containing "0.0" and a spinner.
- Maximum:** A checkbox is unchecked, followed by a text box containing "1.0" and a spinner.
- Default Value:** A text box containing "0.0" and a spinner.

At the bottom of the dialog are four buttons: "OK", "Apply", "Cancel", and a help button (a circle with a question mark).

## 4-Data Input & Data Output (optional)







- Tools can optionally take input data or send a data output or both.
- Use the Workflow Input node to define the incoming data structure and the Workflow Output node to specify the result table the tool produces.
- The agent does not access raw data directly but can trigger tools that process and summarize data as needed.



### Workflow Input

Parameter name

Description







B I U      

This tool takes as input a table with at least one column called Temperature and outputs the Average Temperature of that column.

### Workflow Output

Parameter name

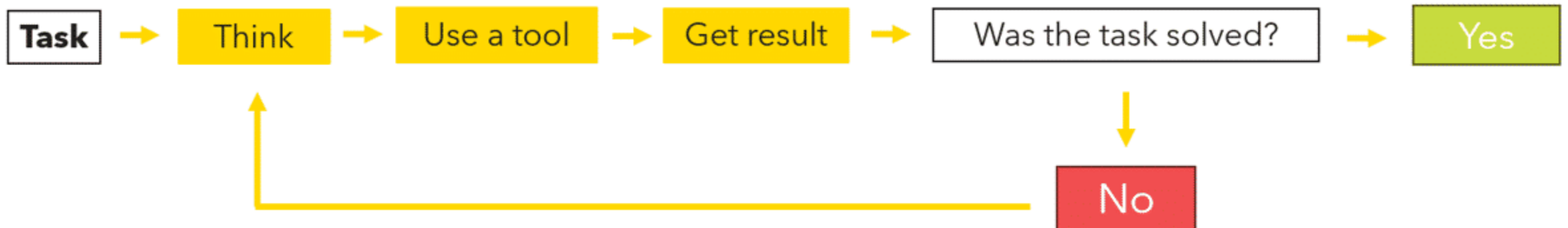
Description

B I U      

This tool outputs the average temperature calculated on the Temperature column. It adds a column called Average Temperature with the calculated value added to it.

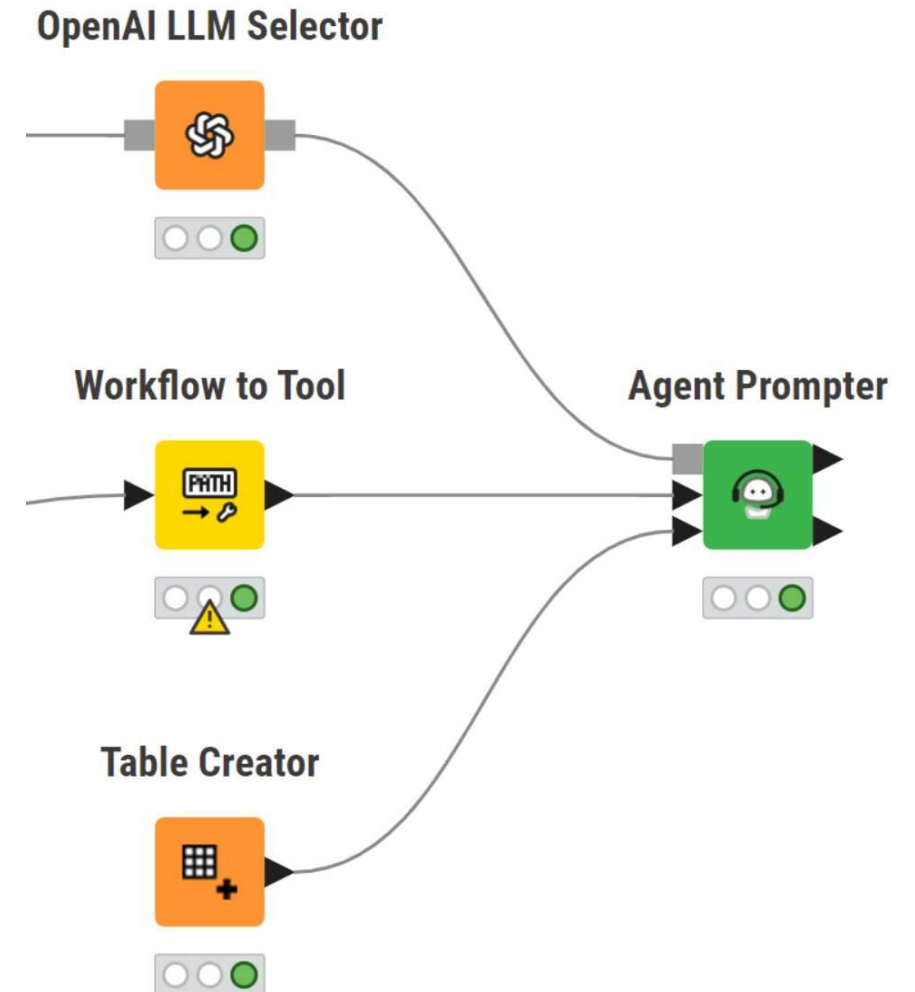
## How does Agents Work?

- AI Agents work by a series of reasoning loop executions.
- In a way, agents are performing trials and errors until the required task is completed.
- Here are the steps in agents' execution:
  - The agent thinks about the task.
  - Chooses whether to call a tool.
  - Evaluates the result produced by the tool.
  - Decides if further steps are needed.
  - Continues until the task is completed.



# Agent Prompter




- In KNIME Analytics Platform, an AI agent is implemented by the Agent Prompter node, executing the loop described above.
- Agent Prompter Inputs:
  1. A connection to the selected LLM Model
  2. List of tools available to the agent
  3. Optional input data if some tools require external data.
- Agent Prompter Outputs:
  1. Result of execution showing interaction between agent and tools.
  2. Optional output data.



# Agent Prompter Configuration Panel




In the configuration panel, we supply the information necessary to execute the Agent Prompter node

**Agent Prompter**

**System message**  **System Message**  

Defines the agent's role, goals, and behavior

You are a virtual assistant. Respond to user queries using the tools at your disposal with a rationale how you got there.

**User message**  **User Message**  

The task or the question to solve

The input table contains information about employees. Retrieve the rations for those employees.



Dialog - 5:8 - Agent Prompter



**System message**

You are a restaurant assistant agent.

Always continue reasoning until the user's request has been fully handled. Use your available tools to verify data and make decisions. Do not guess.

**User message**

Book a table for two people for the 2025.26.6

**Conversation column name**  

Conversation

**Tool column**

Tool

[Show advanced settings](#)

Cancel Ok



## Example

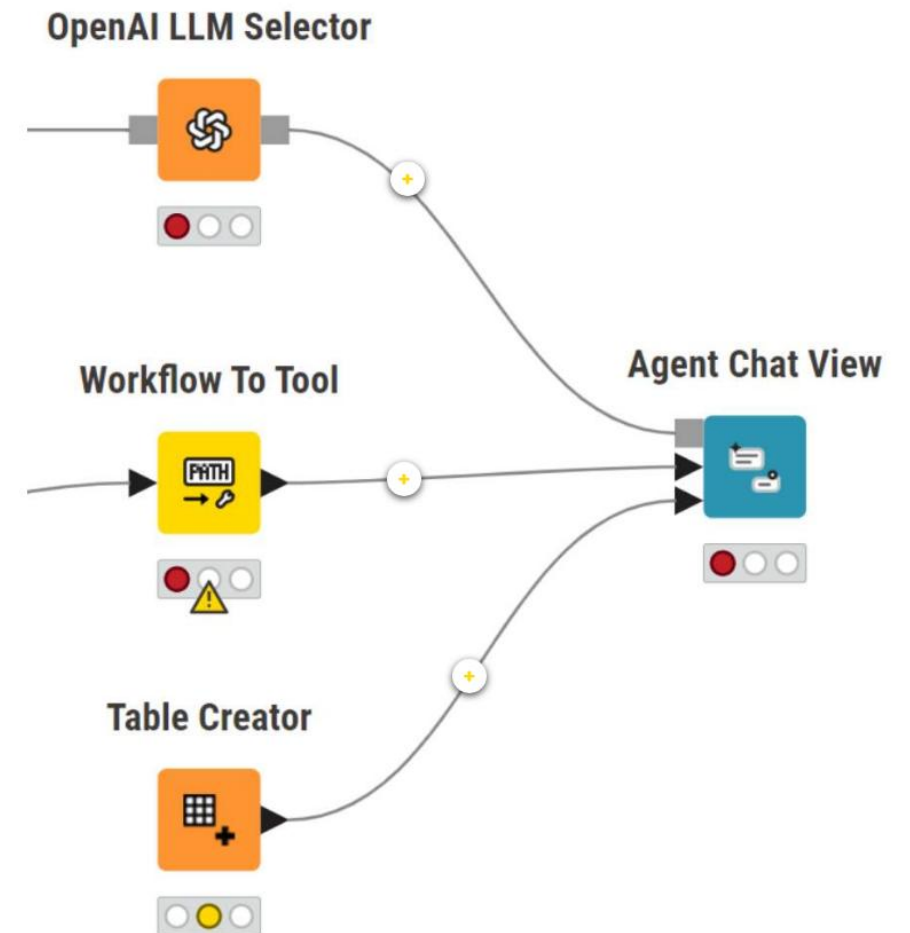
The Agent Prompter then outputs the conversation between the user, the agent, and the tool.



Conversation (Table)				
Rows: 5   Columns: 1				
<input type="checkbox"/>	#	RowID	Conversation Message	
<input type="checkbox"/>	1	Row0	User	
<b>Data Tools Interface</b>				
You have access to tools that can consume and produce data. The interaction with these tools is mediated via a data repository that keeps track of all available data items. The repository is represented as a map from IDs to data items.				
<input type="checkbox"/>	2	Row1	User	
Book a table for two people for the 2025.26.6				
<input type="checkbox"/>	3	Row2	AI	
table_availability				
configuration: booking_date-2: "2025-06-26T00:00:00+02:00[Europe/Berlin]" number_people-4: 2				
<input type="checkbox"/>	4	Row3	table_availability	
The booking for table T1 with 2 people, on 2025-06-26 was confirmed!				
<input type="checkbox"/>	5	Row4	AI	
The table for two people has been successfully booked for June 26, 2025.				

## the Agent Chat View Node

While the Agent Prompter node takes a single prompt and returns the result, the Agent Chat View node provides an interactive chat interface for live agent conversations.



# Agent Chat View Configuration Panel

We supply the information necessary to execute the Agent Chat View node in the configuration panel.

**Agent Chat View**

**System message** ⓘ

## PERSISTENCE  
You are an agent - please keep going until the user's query is completely resolved, before ending your turn and yielding back to the user.

**System Message** < >  
Defines the agent's role, goals and behavior for the conversation

**Tool column** ⓘ

No value selected

**Tool column** < >  
The column holding the tools the agent can use

**Initial message** ⓘ

No value selected

**Initial Message (optional)** < >  
The initial instruction or question by the user

## Agent Chat View

System message ⓘ

## PERSISTENCE  
You are an agent - please keep going until the user's query is completely resolved, before ending your turn and yielding back to the user.

Tool column ⓘ

No value selected

Initial message ⓘ

☐ Show tool calls and results ⓘ

[Show advanced settings](#)

Discard

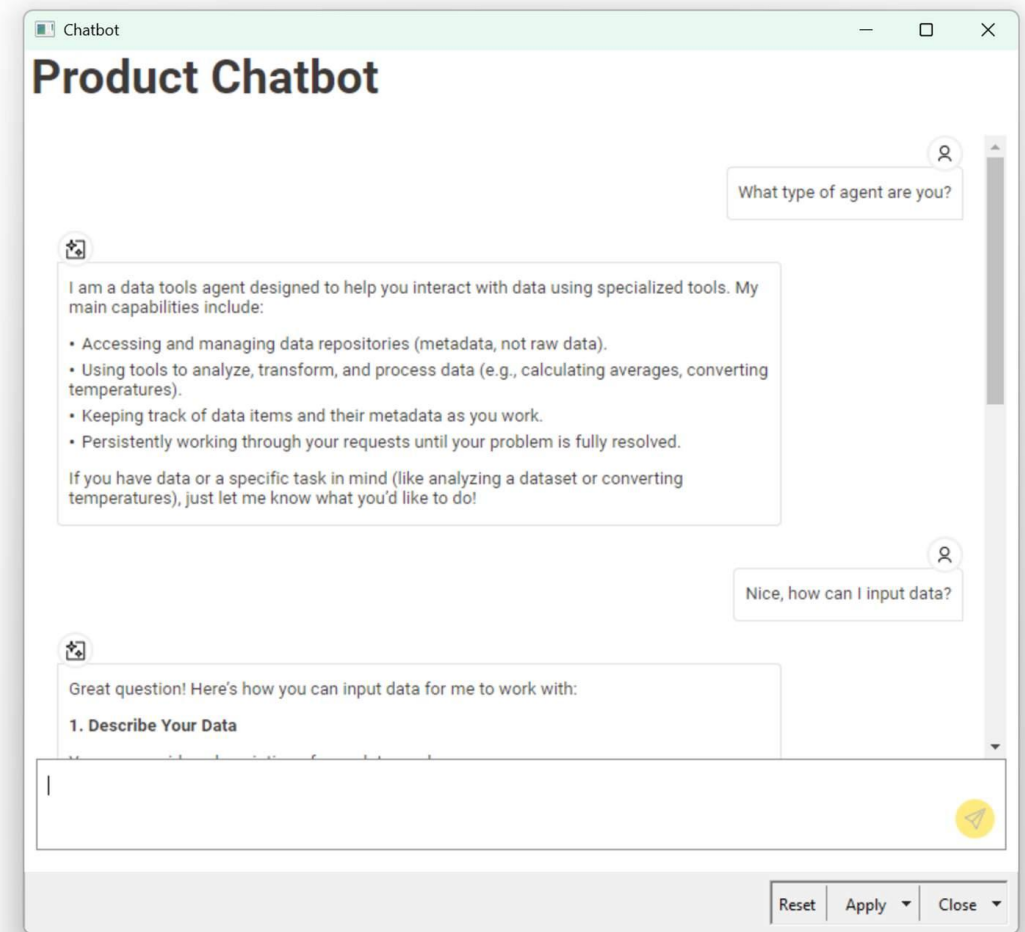
Apply and Execute

Apply

## Example

- During execution, the node opens a chat interface where additional user inputs can be provided interactively.
- The agent reasons, selects tools, processes results, and generates responses in real time.
- This node can be embedded inside a component, allowing it to be deployed and shared via KNIME Business Hub, making the agent available as an interactive application to end users.

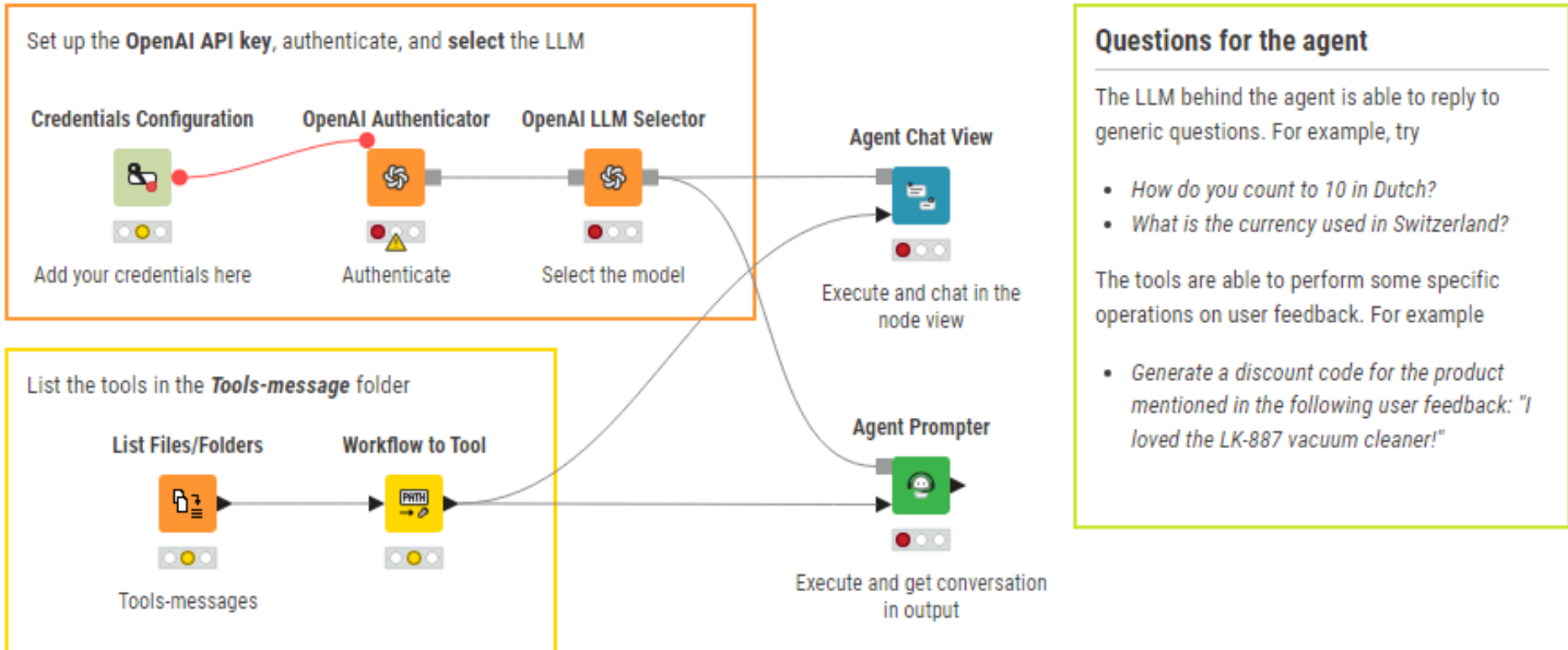
Chatbot



# 01 Simple Agent

This workflow demonstrates how to create an agent that is able to analyze product feedback and generate discount codes using two tools. Two solutions are showed:

- The **Agent Chat View** creates an interactive view where you can directly prompt the agent.
- The **Agent Prompter** prompts the agent with one user question and produces the conversation in output.



This workflow can be downloaded as following:

1. Download Course Workflows from VClass
2. Goto Generative AI Folder -> Getting Started Courses -> Data Aware Agentic AI – Getting Started
3. Open 01 Simple Agent

# Example Workflows



KNIME Learning Center

## Download the example workflows

This course does not include exercises. To earn your microcredentials, you'll need to complete a knowledge check at the end.

You can, however, download the example workflows featured throughout the course and explore them to apply the knowledge you have acquired.

Download the **example workflows** from the KNIME Community Hub.

DOWNLOAD

